2021 17th International Conference on Mobility, Sensing and Networking (MSN) | 978-1-6654-0668-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/MSN53354.2021.00114

2prong: Adaptive Video Streaming with DNN and MPC

Yipeng Wang, Tongqing Zhou, Changsheng Hou, Bingnan Hou, Zhiping Cai
College of Computer, National University of Defense Technology, Changsha, China
Email: wangyipeng, zhoutongqing, zpcai@nudt.edu.cn

***Abstract*—Adaptive bitrate (ABR) algorithms are often used to optimize the quality of user experience (QoE) during video playback. In the client-side video player, the buffer size and predicted throughput are mainly used to improve user's QoE. However, due to the randomness of mobile network traffic and the heavy-tail effect of the network, it is very difficult to predict throughput. We innovatively use Bayesian neural network to dynamically evaluate video signals. Unlike previous neural network solutions, we use probability distributions instead of point estimates to predict throughput, which can effectively evaluate QoE metrics. Our contributions are to first (i) use of Bayesian neural network to guide video adaptive bitrate adaptation, and then (ii) propose a bitrate adaptive algorithm denoted 2prong, which utilizes high-dimensional contextual information such as buffer occupancy, predicted throughput and video quality to find the most valuable information for quality adaption in real time. We demonstrate the effectiveness of the 2prong algorithm using a simulation testbed. By comparing with other methods, it is demonstrated that 2prong can improve the video quality of video streaming transmission.**

***Index Terms*—video streaming, bitrate adaptation, model predictive control, Bayesian network**

I. Introduction

According to Cisco VNI [1], video traffic will account for over 82% of all IP traffic by 2022, video applications have become the absolute dominant application. While video traffic continues to grow, users also need better video quality. In recent years, there have been many technologies that can improve the quality of user experience, but the core of these technologies is the adaptive bitrate(ABR) algorithm, which dynamically determines the quality of each segment. The ABR algorithm needs to strike a balance between high quality, minimal rebuffering and few quality switches to maximize the overall QoE [2], [3], [4]. A lot of work on internet video streaming has been devoted to the design of better ABR algorithms [2], [3], [4], [5], [6], [7], [8], and further improvements are still desired.

For relatively long-term video streaming, how does the ABR algorithm predict future network and client conditions will directly affect QoE. The original algorithm is generally based solely on buffer or throughput, but because of the single indicator, it is impossible to correctly evaluate the client and network conditions at this time, and can not choose the optimal bitrate. Most of the state-of-the-art ABR algorithms are designed to work with the neural network. However, due to the neural network algorithm requires more resources, such predictions can't correctly reflect the bitrate [2], [9], so the ABR algorithm may not get the most balanced QoE. To address the limitations of neural network, we use model predictive control (MPC) to improve the overall QoE [10]. In contrast to neural network, MPC can directly combine both

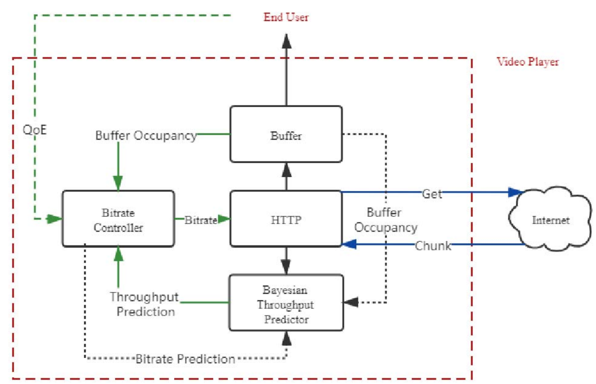

Fig. 1. Overview of 2prong.

rate-based and buffer-based feedback signals. High quality versions can be obtained from lower quality ones by scheduling through the client [11]. MPC can download suitable segments for the network status at this time, and reduce the rebuffering when the network status changes suddenly. If the network condition is better, it can be upgraded to a higher quality segment. MPC can eliminate network fluctuation, increase flexibility, and improve QoE. [12].

As shown in Figure 1, 2prong mainly uses the available bandwidth, playback buffer size and QoE variations to adjust the video bitrate. In predicting the throughput, we also utilize the buffer and QoE values so that the bitrate of the next video block can be predicted more accurately. The algorithm must avoid video streaming problems such as video instability, quality oscillations, and buffer starvation, while improving the viewer's QoE. The algorithm needs to strive to achieve (i) maintaining the number of video blocks in the playback buffer and avoiding re-buffering, (ii) minimal start-up latency in the case of real-time video playback, (iii) making full use of network bandwidth resources, and (iv) avoiding the frequent switching caused by video bitrate oscillations caused by frequent switching.

The network information obtained through MPC can be used as an available feature vector for video quality adaptation. We model the client's decision on a video segment quality as a model predictive control aiming to optimize an objective QoE metric that comprises (i) the General latency(gl), (ii) the Startup latency (sl), and (iii) the buffering time (bt) [14].

Our contributions in this paper can be summarized as: (1) For the first time, we use Bayesian neural network to guide

978-1-6654-0668-0/21/$31.00 ©2021 IEEE
DOI 10.1109/MSN53354.2021.00114

video bitrate adaptation, which is a huge improvement to the client-Assisted Adaptation method. (2) We propose 2prong, a Bayesian method for designing ABR algorithms that fully improve mobile network video quality. (3) We show emulation testbed results and demonstrate the fundamental differences to the established state-of-the-art quality adaptation algorithms.

The remainder of this paper is organized as follows: In Sect. II, we review relevant related work on client-assisted adaptation and Bayesian neural network. In Sect. III, we present the relevant background and working mode on Bayesian neural network. In Sect. IV, we model the quality adaptation problem, show how ABR streaming uses 2prong and define a weighted reward based on QoE metrics. We describe the evaluation testbed before providing emulation results in Sect. V. Section VI concludes the paper.

II. Related work

In previous papers, many HTTP Adaptive Streaming (HAS) systems can select and control the video bitrate on the client, server and network respectively. Client selects the bitrate based primarily on the dynamic size of the buffer and the end-to-end bandwidth available.

Client-based bitrate adaptation can be divided into three categories: (1) available bandwidth-based, (2) playback buffer-based, and (3) hybrid. Several solutions have been proposed recently (e.g., [16], [17], [18], [19], [20]). The rate-based approach selects the highest bitrate based on the estimated future available throughput. However, previous work has pointed out that HTTP throughput estimation can be highly biased [21], which leads to the ineffectiveness of traditional rate-based approaches. Some solutions try to reduce the bias by smoothing the throughput estimation [22] or by choosing a better scheduling strategy [20]. On the other hand, recent work has made more use of buffer-based algorithms [19]. Instead of using throughput estimation, such algorithms use buffer occupancy as a feedback signal and always keep the buffer occupancy rate at an ideal level, basically discarding the throughput information.

However, neither scheme can control the bitrate adaption well, so a hybrid adaption model is created.Both Model Predictive Control (MPC) and RobustMPC [4] all belong to this category. It consists of two modules: a throughput predictor that predicts the available bandwidth in the future and a buffer predictor that informs how much buffer the prediction model will occupy over time. QoE is affected by both throughput and buffer, and determines the future video bitrate based on the quality and size of the acquired chunks.MPC uses the model to plan in a limited range (e.g., the next 5-8 chunks) a sequence of chunks to maximize the expected QoE.

CS2P and Oboe-tuned RobustMPC are improvements over traditional MPC; they utilize better throughput prediction models, but still provide buffer and QoE information in the same way as traditional MPC. the throughput predictors of CS2P and Oboe-tuned RobustMPC are trained using simulated real datasets that record the changes in throughput during streaming. CS2P observes that clients are similar and clusters similar clients. For the changing throughput, it converts continuous to discrete and uses Markov process to describe the change of throughput; Oboe focuses on the change of network link state, when the link state changes, the algorithm must immediately obtain the buffer situation of the client at this time and observe the buffer state, and adjust the video bit rate if necessary to improve the user's viewing quality.

Fugu is also an improvement on the traditional MPC algorithm. The throughput predictor is very different from the previous one: it changes the throughput predictor to a Transmission Time Predictor (TTP), which directly predicts the transmission time of a chunk in the future, instead of predicting the network throughput metric. The throughput predictor models the transmission time of a chunk linearly proportional to its size, but it is known that the throughput size varies with the file size due to the effect of congestion control and the different network link capacities while video data chunks pass through the network.

Pensieve [2] is also an ABR scheme based on a deep neural network model. However, rather than simply using a neural network to make predictions, Pensieve directly decides which chunks to send. While CS2P and Fugu's TTP, although trained using supervised learning methods, cannot train a scheme that can make direct decisions because it requires not only data, but also a training environment and the need to judge the consequences of reacting to a series of decisions. This is called reinforcement learning (RL). In general, reinforcement learning must have a training environment and a testing environment to test as many situations and actions as possible to improve the algorithm effectiveness. If a performance metric is changed, there must be a significant change in the algorithm effect. Therefore, reinforcement learning algorithms generally perform poorly in systems with much noise.

III. Overview

In this section, we design a model predictive control (MPC) method that can combine the throughput size and the buffer occupancy rate for bitrate adaptation on the client. In order to deal with the problem of inaccurate throughput prediction under highly variable network conditions, we have also added a predictive neural network.

A. Why MPC?

First of all, model predictive control may be naturally suitable for the problem of bitrate adaptation. But we cannot say that MPC is the best choice among all possible bitrate control algorithms. We can only say that the MPC algorithm is more effective in mobile network video streaming.

Ideally, given a video $[t_k, t_{k+1}]$, , the QoE optimization problem can be directly calculated by the optimal bitrate R_1,... , R_k and the start-up delay T. In practice, however, we do not have access to this perfect information, making it difficult to optimize the optimal solution offline. Although perfect information for the future as a whole is not available, it is possible to obtain reasonably accurate throughput predictions over a short span of $[t_k, t_{k+N}]$. In the future. This is because the network conditions are quite stable over a short period of time and do not change drastically within a few tens of seconds [25]. So we can use this feature of throughput to run QoE optimization by applying the first bitrate R_k and moving the horizon forward to $[t_k, t_{k+1}]$. This scheme is known as model predictive control (MPC) [26]. The advantage of MPC is that MPC can use prediction to optimize complex control objectives in dynamic systems online subject to constraints.

B. Robust MPC

The most basic MPC algorithm assumes that throughput can be accurately predicted. However, under the harsh network conditions such as cellular networks, it is not possible to predict throughput accurately. If the throughput metric is always overestimated, rebuffering may be induced. To offset the prediction error, [4] develop a robust MPC algorithm.

Robust MPC can only solve the QoE balance problem in the worst case of the network. Robust MPC does not treat the throughput as a point estimate $\hat{C}_t$, but assumes that the actual throughput can take any value in the range $\left\{\hat{C}_t, t \in [t_k, t_{k+N}]\right\}$. But this exposes a shortcoming: Robust MPC is more conservative. In the video bitrate selection, this algorithm always chooses according to the lower limit of the predicted throughput. So in our work, we use the maximum prediction error of the last few chunks as the bound, which can effectively avoid this problem.

C. Why Bayesian neural network?

Deep neural networks (DNNs) do not need to understand the task in advance, but use examples to understand the task, which can be effectively reused. [2]. They can easily scale to millions of data points and can be optimized by stochastic gradient descent, so pensieve has used DNNs for bit-rate adaptation with good results. However, the current neural network architecture lacks a measure of predictive uncertainty, which Bayesian neural networks incorporate. Bayesian neural networks perform a posterior inference through parameters, thus preventing overfitting.

There are many ways to construct Bayesian neural networks. However, in this paper, we use Backprop's Bayes method to construct a Bayesian DNN. It is difficult to obtain accurate weight values of neural networks because Bayesian neural networks involve too many parameters and it is also impossible to cluster the parameters. Therefore we will approximate the intractable true posterior estimated probability distribution with the probability distribution with variance, which conforms to the properties of the Gaussian distributions denoted as the probability distribution. The shape of Gaussian variational posterior distributions is determined by their variance, expressing an uncertainty in the estimation of the parameters for each model.

IV. Framework

In this section, we describe the design of 2prong. We apply Bayesian neural network to model predictive control, and apply the algorithm to video streaming sessions. We first have to design a bitrate adaptation algorithm to select the best bitrate video chunk version. R_K is the video bitrate chosen by the algorithm and the client buffer occupancy can be expressed as : $B_{k+1} = [(B_k - r_k L/C_k) + L]$, B_K is the client buffer occupancy when the k-th video chunk is downloaded, $r_k L/C_k$ denotes the time required to download the k-th video chunk, and C_k is the average network throughput of the network at this time.

A. Bayesian neural network algorithm

We now describe our Bayesian neural network algorithm.

We use Bayesian neural networks (BNNs) to predict throughput, sampling throughput through a Bayesian posterior probability distribution. Bayesian neural networks (BNNs) are uncertain due to the lack of sufficient data samples to determine the dynamically true probability distribution. To normalize the uncertainty, we determine only a small range of network throughput, rather than a definite point. We assume that the probability of network throughput follows a Gaussian distribution that varies with time. Thus, using a BNN-based predictor, we can predict the future throughput, taking into account both epistemic uncertainty and evasive uncertainty.

Inputs: After the download of each chunk t, state S_t= (X_t,N_t,B_t) will be input to the Bayesian neural networks. X_t is the network throughput measurements for the past k video chunks; N_t is a vector of m available sizes for the next video chunk; B_t is the current buffer level.

Policy: Upon receiving S_t, 2prong needs to take an action A_t corresponding to the bitrate of the next video chunk. It selects actions based on a policy, defined as a probability distribution over actions $\pi : \pi(S_t, A_t) \rightarrow [1, 1]$. $\pi(S_t, A_t)$ is the probability that action At is taken in state S_t. 2prong uses Bayesian neural network (BNN) to represent the policy with a manageable number of adjustable parameters, θ, which we refer to as policy parameters. Using θ, we can represent the policy as $\pi\theta(S_t, A_t)$. BNNs have recently been applied successfully to solve large-scale RL tasks, and BNN can directly observe the signal without manually adding features.

Policy gradient training: After applying each action, the simulated environment provides the BNN learning agent with a reward R_t for that chunk. BNN hopes to get the biggest reward. Thus, the reward must reflect the performance of each chunk download to optimize specific QoE metrics.

B. Bayesian model predictive control

MPC performs the following three key steps: predict, optimize, and apply.

1) Predict: We use Bayesian neural network to predict throughput, improving the accuracy of throughput prediction will increase QoE's revenue. In other words, MPC can have the robustness of throughput prediction.
2) Optimize: This is the core of the MPC algorithm: Given the current buffer occupancy B_k, previous bitrate R_{k-1} and throughput prediction$C_{[t_k,t_{k-1}]}$, find optimal

bitrate R_k. $R_k = f_{mpc}(R_{k-1}, B_k, C_{[t_k,t_{k-1}]})$, implemented by solving QOE_MAX_STEADY.In the start-up phase, it also optimizes start-up time T_s as $[R_k, T_s]$ = $f^{st}_{mpc}(R_{k-1}, B_k, C_{[t_k,t_{k-1}]})$,implemented by solving QOE_MAX.
3) Apply: Start to download chunk k with R_k and move the horizon forward. If the player is in start-up phase, wait for T_s before starting playback.

2prong's throughput prediction is uncertain (the variance of the throughput prediction indicates how accurate the algorithm thinks the predicted network throughput is), which in practice leads to a conservative MPC policy that tries to find an optimal bitrate so that rebuffering errors do not occur. When the user ends up choosing the video chunk with the best bitrate recommended by the algorithm, it ensures that as few rebuffering errors as possible occur, even if the throughput prediction is not as accurate.

Our 2prong algorithm has a relatively large advantage over RobustMPC. Because the robustness of RobustMPC depends mainly on the magnitude of the error in previous throughput predictions, it essentially amounts to an inaccurate estimate that uses the prediction uncertainty from past sampling to approximate the current prediction uncertainty. In contrast, the uncertainty of 2prong estimation comes from the current forecasting process. In practice, the 2prong algorithm also performs very well.

At the same time, due to the limitations of mobile devices, our algorithm must be lightweight. The high computational overhead of MPC is a very big test for low-end mobile devices, which will be the main video clients in the future. Since the MPC algorithm has to start functioning before the player starts downloading each chunk, once the algorithm time overhead is too high it will negatively affect the QoE of the client.

V. Evaluation

In this section, we evaluate 2prong through experiments. Our experiments cover a wide range of network conditions, but mainly focuses on the mobile network environment. Our results show that 2prong can match or exceed the best available solutions in mobile network scenarios, and the average QoE improvement range is 7.4%–12.5%.

A. Methodology

1) Network traces: FCC and 3G/HSDPA are two public datasets. We use these two datasets to simulate real network conditions and evaluate 2prong and the most advanced ABR algorithm. The FCC dataset mainly records throughput. There are a total of more than 1 million throughput tracking records, and each tracking record records more than 400 average throughputs at a granularity of 5 seconds. We randomly selected the "Web browsing" category in the August 2016 data and generated 1,000 tracking records, each of which lasted 320 seconds. The HSDPA dataset is the data obtained when the mobile device is streaming video during the transmission process (for example, via a bus, train, etc.). In order to be consistent with the FCC dataset, we have generated 1000 HSDPA tracking records (each span is 320 seconds). To avoid scenarios where the network cannot support any available bitrate for an extended period, we only considered original traces whose average throughput is less than 6 Mbps, and whose minimum throughput is above 0.2 Mbps.

2) Adaptation algorithms: We compare 2prong with the following latest technology algorithms that represent bitrate adaptation:

1) Buffer-Based (BB): is a purely buffer-based algorithm. This algorithm maintains a storage area of 5 seconds and a buffer of 10 seconds. The core is to always keep the buffer occupancy rate above 5 seconds, and immediately reduce the bitrate once it is less than 5 seconds. If the buffer occupies more than 15 seconds, the algorithm selects the highest bitrate.
2) BOLA: is also a purely buffer-based algorithm, but using Lyapunov optimization to select the bitrate, and is theoretically the best purely buffer-based algorithm.
3) Rate-based (RB): It is a purely throughput-based algorithm that mainly uses the harmonic mean of network throughput in the past 5 chunks to predict the throughput when downloading the next chunk, and then selects the highest throughput that is lower than the predicted throughput available bitrate.
4) RobustMPC: is a development of MPC algorithm and the main idea is to infinitely approximate the throughput prediction value to the lower limit and calculate the error between the predicted value and the true value.
5) Pensieve: base on a deep neural network. Pensieve uses neural networks to predict throughput and buffers, and also uses neural networks to determine how many bitrate video chunks to send in the future.

B. QoE metrics

Factors affecting QoE are generally classified into (a) perceptual factors, which can be directly perceived by viewers, and (b) technical factors, which can indirectly affect QoE. Perceptual factors include video image quality, video initial delay, video mid-pause time, pause frequency, and video quality switching amplitude and switching frequency. These competent factors can have different weights depending on the subjectivity of users. Several studies have shown that most users perceive that video pauses in the middle of a video have a more critical impact on QoE than the initial slight delay, that longer pauses during video playback reduce user-perceived quality, and that frequent changes in video quality can also have a large negative impact on QoE. The technical factors that affect QoE are the algorithms, parameters, and hardware/software used in the video streaming delivery system. Specifically, these factors include the encoding parameters, video quality and video clip size on the server side, and the selection logic, device capabilities and video content on the client side. Both perceptual and technical factors are realistic issues to consider, and QoE must strike the best balance between conflicting goals (e.g., minimum number of pauses vs. high quality bitrate) to improve viewer satisfaction.

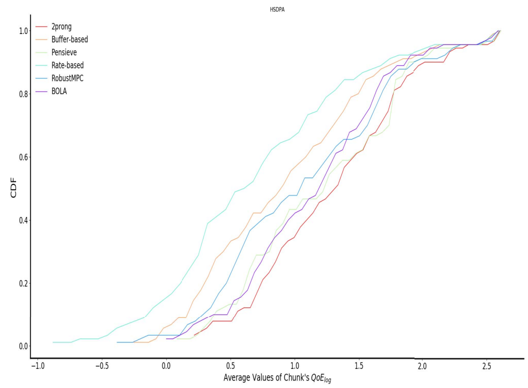

Fig. 2. Comparing 2prong with ABR algorithms on the HSDPA broadband dataset

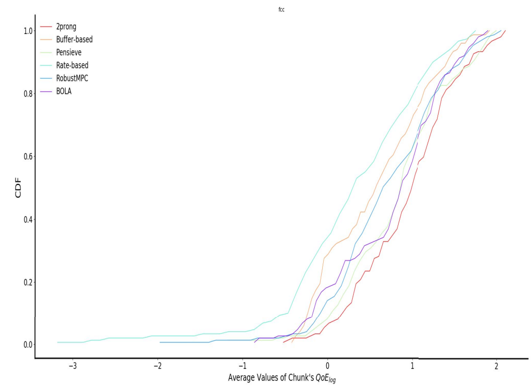

Fig. 3. Comparing 2prong with ABR algorithms on the FCC broadband dataset

Different users have different preferences of QoE indicators for video streaming. Therefore, we must integrate various QoE indicators. Through observation, we have selected the following indicators:

$$QoE = \sum_{n=1}^{N} q(B_k) - \mu \sum_{n=1}^{N} q(r_k L/C_k) - \sum_{n=1}^{N} |q(B_{k+1}) - q(B_k)|$$

for a video with N chunks. B_k represents the bitrate of $chunk_n$ and $q(B_k)$ maps that bitrate to the quality perceived by a user.$r_k L/C_k$ represents the rebuffering time that results from downloading $chunk_n$ at bitrate B_k, while the final term penalizes changes in video quality to favor smoothness.

C. 2prong vs. Existing ABR algorithms

We compared 2prong with five state-of-the-art ABR algorithms. We use the dataset described in §5.1 to evaluate the quality of six ABR algorithms including 2prong. Figure 2 shows the average QoE of each algorithm on the HSDPA dataset, and Figure 3 shows the average QoE of each algorithm on the FCC dataset.

From these results, we found two interesting phenomena. First, 2prong meets or exceeds five other ABR algorithms in terms of total QoE metrics, with the closest to 2prong being pensieve; this illustrates the importance of neural networks. Second, On HSDPA dataset, the average QoE of 2prong is about 9.5% higher than pensieve. On FCC dataset, the gap between Pensieve and 2prong has widened to about 12.5%, which shows that the 2prong algorithm has more advantages in mobile networks.

Figures 4 show the average values of the three separate QoE metrics on the FCC and HSDPA datasets. We find that 2prong guarantees the best QoE because the bitrate is maximized, while the penalty of smoothness and rebuffering is less. The results for Pensieve and 2prong are similar. In addition, since the throughput predicted by the RobustMPC algorithm is consistently low, there is very little video rebuffering, although the penalty of smoothness is larger. In contrast, the video bitrate in the Bola and Buffer-based algorithms, however, fluctuates more, and even though good performance is achieved in terms of bitrate utility and penalty for regression, the smoothness penalty is extremely high and the optimal QoE cannot be achieved.

Figure 4 uses CDF to illustrate the relative pros and cons of all the QoE indicators, mainly in terms of the utility of the average playback bitrate, the rebuffering penalty, and the smoothness penalty. As shown in figure 4,The advantage of 2prong is that it can maintain a relatively stable bitrate. 2prong is used to handle fluctuations in network throughput, thereby reducing rebuffering. In addition, Figure 4 also shows that 2prong is not superior to all the most advanced solutions in every QoE factor, and there is room for improvement.

VI. Conclusion

In this paper, we explored the design of ABR algorithms for Mobile network video streaming through Bayesian model predictive control-based approach 2prong. In contrast to state-of-art adaptation algorithms, we take Uncertainty-Aware Robust Adaptive video streaming transmission. In this setting, streaming context information includes neural network indicators obtained through QoE metrics. We evaluated the performance of our proposed 2prong on an emulated streaming system, with commonly used video.Over a broad set of network conditions and QoE metrics, we found that 2prong outperformed existing ABR algorithms by 7.4%–12.5%.

VII. Acknowledgment

This work is supported by the National Natural Science Foundation of China (62072465, 62172155, 62102425, 62102429) and the Hunan Provin-cial Postdoctoral Innovation Talent support Program (No. 2021RC2071).

References

[1] 2019. Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper. https://www.cisco.com/c/en/us/solutions/collateral/service-provider/ visual-networking-index-vni/white-paper-c11-741490.html

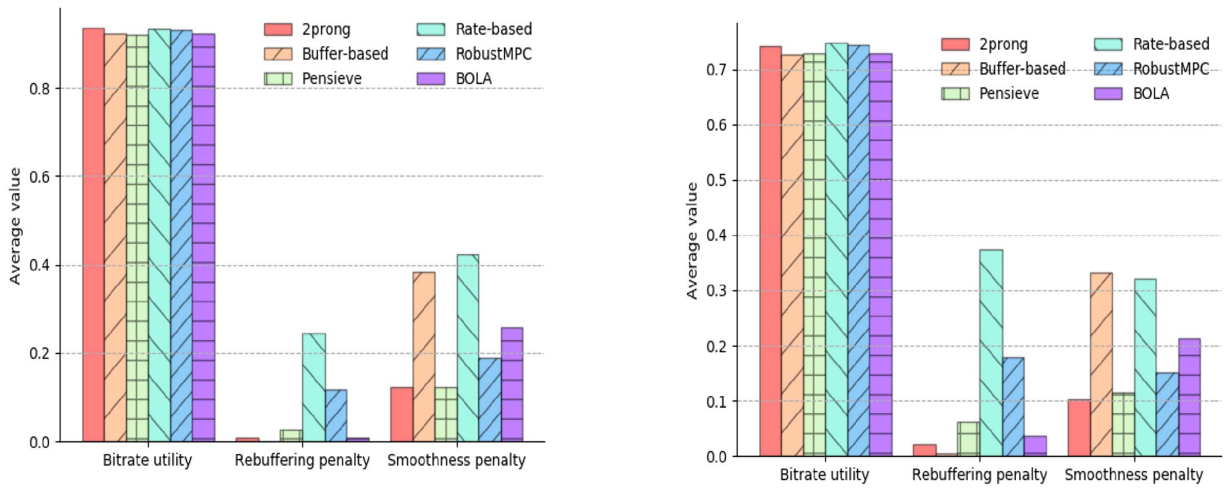

Fig. 4. Comparing 2prong with ABR algorithms by analyzing their performance on the individual components in the general QoE definition with HSDPA and FCC dataset

[2] Hongzi Mao, Ravi Netravali, and Mohammad Alizadeh. 2017. Neural Adaptive Video Streaming with Pensieve. In Proceedings of the Conference of the ACM Special Interest Group on Data Communication. 197–210.

[3] Afshin Taghavi Nasrabadi and Ravi Prakash. 2018. Layer-Assisted Adaptive Video Streaming. In Proceedings of the 28th ACM SIGMM Workshop on Network and Operating Systems Support for Digital Audio and Video. 31–36.

[4] Xiaoqi Yin, Abhishek Jindal, Vyas Sekar, and Bruno Sinopoli. 2015. A Control Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. In Acm Conference on Special Interest Group on Data Communication. 325–338.

[5] K. Spiteri, R. Urgaonkar, and R. K. Sitaraman. 2016. BOLA: Near-optimal bitrate adaptation for online videos. In IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications. 1–9.

[6] Federico Chiariotti, Stefano D'Aronco, Laura Toni, and Pascal Frossard. 2016. Online Learning Adaptation Strategy for DASH Clients. In Proceedings of the 7th International Conference on Multimedia Systems. Article 8, 12 pages.

[7] Maxim Claeys, Steven Latré, Jeroen Famaey, Tingyao Wu, Werner Van Leekwijck, and Filip De Turck. 2013. Design of a Q-Learning based client quality selection algorithm for HTTP Adaptive Video Streaming.

[8] Maxim Claeys, Steven Latré, Jeroen Famaey, TY Wu, W Van Leekwijck, and Filip De Turck. 2014. Design and optimisation of a (FA)Q-learning-based HTTP adaptive streaming client. CONNECTION SCIENCE 26, 1 (2014), 22.

[9] Jaehyun Hwang, Junghwan Lee, and Chuck Yoo. 2016. Eliminating bandwidth estimation from adaptive video streaming in wireless networks. Signal Processing: Image Communication 47 (2016), 242 – 251.

[10] E. F. Camacho and C. B. Alba. Model Predictive Control. Springer, 2013.

[11] Farzad T ashtarian, Alireza Erfanian, and Amir Varasteh.2019.S2V C: An SDN-based Framework for Maximizing QoE in SVC-Based HTTP Adaptive Streaming.

[12] Grad: Learning for Overhead-aware Adaptive Video Streaming with Scalable Video Coding,Liu, Yunzhuo (Shanghai Jiao Tong University, Shanghai, China); Jiang, Bo; Guo, Tian; Sitaraman, Ramesh K.; Towsley, Don; Wang, Xinbing.MM 2020 - Proceedings of the 28th ACM International Conference on Multimedia.

[13] T. Stockhammer, "Dynamic Adaptive Streaming over HTTP –: Standards and Design Principles," in Proceedings of the Second Annual ACM Conference on Multimedia Systems, ser. MMSys '11. New Y ork, NY , USA: ACM, 2011, pp. 133–144. [Online]. Available:http://doi.acm.org/10.1145/1943552.1943572.

[14] Rldish: Edge-Assisted QoE Optimization of HTTP Live Streaming with Reinforcement Learning Wang, Huan (University of Victoria, Department of Computer Science, Canada); Wu, Kui; Wang, Jianping; Tang, Guoming Source: Proceedings - IEEE INFOCOM, v 2020-July, p 706-715.

[15] CBA: Contextual Quality Adaptation for Adaptive Bitrate Video Streaming Alt, Bastian (Bioinspired Communication Systems Lab (BCS), Technische Universit Darmstadt,Germany); Ballard, Trevor; Steinmetz, Ralf; Koeppl, Heinz; Rizk, Amr Source: Proceedings - IEEE INFOCOM, v 2019-April, p 1000-1008.

[16] G. Tian and Y . Li. Towards Agile and Smooth Video Adaption in Dynamic HTTP Streaming . In Proc. CoNext, 2012.

[17] J. Jiang, V . Sekar, and H. Zhang. Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE. In Proc. CoNext, 2012.

[18] S. Akhshabi, L. Ananthakrishnan, A. Begen, and C. Dovrolis. Server-Based Traffic Shaping for Stabilizing Oscillating Adaptive Streaming Players. In Proc. ACM SIGMM NOSSDAV, 2013.

[19] T.-Y . Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson. A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. In Proc. ACM SIGCOMM, 2014.

[20] Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. Begen, and D. Oran. Probe and Adapt: Rate Adaptation for HTTP Video Streaming at Scale. Selected Areas in Communications, IEEE Journal on, 32(4):719–733, 2014.

[21] T.-Y . Huang, N. Handigol, B. Heller, N. McKeown, and R. Johari. Confused, Timid, and Unstable: Picking a Video Streaming Rate is Hard. In Proc. IMC, 2012.

[22] G. Tian and Y . Li. Towards Agile and Smooth Video Adaption in Dynamic HTTP Streaming . In Proc. CoNext, 2012.

[23] G. F. Franklin, J. D. Powell, and M. L. Workman. Digital Control of Dynamic Systems, volume 3. Addison-wesley Menlo Park, 1998.

[24] D. P . Bertsekas, D. P . Bertsekas, D. P . Bertsekas, and D. P . Bertsekas. Dynamic Programming and Optimal Control, volume 1. Athena Scientific Belmont, MA, 1995.

[25] Y . Zhang and N. Duffield. On the Constancy of Internet Path Properties. In IMW, 2001.

[26] E. F. Camacho and C. B. Alba. Model Predictive Control. Springer, 2013.